

CCH1A4 / Dasar Algoritma & Pemrograman

Yuliant Sibaroni M.T, Abdurahman Baizal M.Kom

KK Modeling and Computational Experiment



Tabel

- ▶ Pendahuluan
- ▶ Deklarasi Tabel
- ▶ Pengaksesan Tabel
- ▶ Program dengan Tabel
- ▶ Program dengan Tabel 2 dimensi

Pendahuluan

Tipe data tabel

Dalam banyak bahasa pemrograman, dikenal tipe data terstruktur. Dalam tipe terstruktur setiap variabel dapat menyimpan lebih dari satu nilai data. Pada umumnya macam tipe data terstruktur yaitu **tabel/array**, **record** dan **file**

Definisi

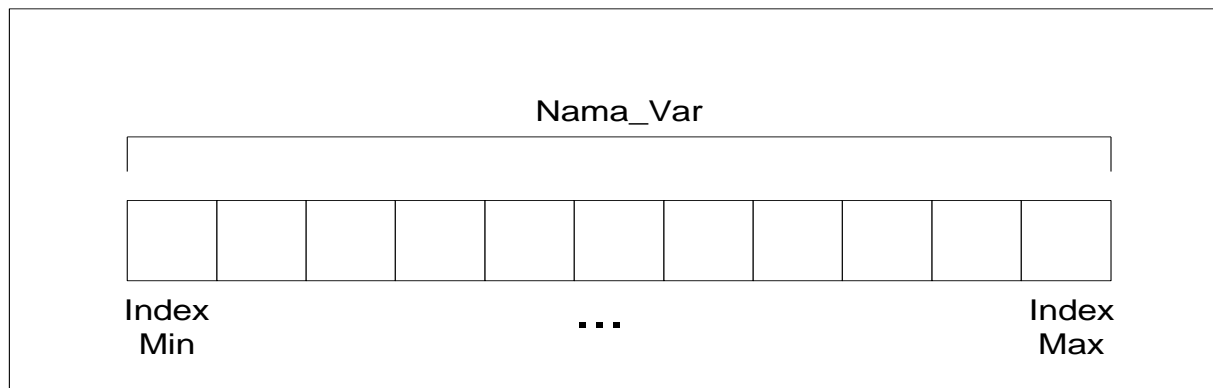
Tabel adalah tipe terstruktur yang menyimpan komponen data yang bertipe sama. Posisi masing-masing komponen dalam tabel dinyatakan sebagai nomor indeks. Komponen tabel dapat diakses melalui indeks.

Tabel

Deklarasi tabel

Indeks harus berupa struktur data ordinal, yaitu yang memiliki keterurutan, misal konstanta integer atau konstanta char

```
Nama_var : array [indeks_min..indeks_maks] of tipe_data
```



Deklarasi Tabel

Contoh 9.1

TabMahasiswa : array [1..20] of string

TabMahasiswa akan digunakan untuk menyimpan 20 nama-nama mahasiswa

Contoh 9.2

TabCountHuruf : array ['a'..'z'] of integer

TabCountHuruf akan digunakan untuk menyimpan jumlah kemunculan huruf-huruf tertentu dari a s.d. z

Pengaksesan Tabel

Cara Pengaksesan

Elemen tabel diakses mengacu melalui **indeks** sebagai berikut:

```
Nama_Var[indeks]
```

Contoh 9.3

```
X ← TabMahasiswa[7]
```

Variabel X akan diisi dengan nama mahasiswa yang tersimpan pada indeks ke-7 pada tabel TabMahasiswa

Pengaksesan Tabel

Contoh 9.4

```
Y ← TabCountHuruf[ 'd' ]
```

Variabel Y akan berisi nilai bilangan bulat yang tersimpan pada indeks ke-4 (d: memiliki indeks 4 karena dimulai dari a) pada array **TabCountHuruf**

Contoh 9.5

```
For i ← 1 to 7 do  
    Output (TabNamaHari [i])
```

Statement program diatas akan mengeluarkan output semua nama hari yang tersimpan pada tabel **TabNamaHari**

Program Dengan Tabel

Contoh 9.6

Proses input (pengisian) elemen tabel

Program Pengisian_Tabel

{Mengisi elemen array A[1..N] dengan harga dari piranti masukan.

IS : sembarang

FS : seluruh elemen array berisi harga yang dibaca dari piranti masukan}

Kamus

A : array [1..50] of integer

i : integer

Algoritma

For i ← 1 to 6 do

Input(A[i])

output('Data ke-3 = ',A[3])

output('Data ke-5 = ',A[5])

Program Dengan Tabel

Contoh 9.6

Hasil running program

```
4
5
3
12
7
8
Data ke-3 = 3
Data ke-5 = 7
```

```
16
5
10
18
32
9
Data ke-3 = 10
Data ke-5 = 32
```

Program Dengan Tabel

Contoh 9.7

Diketahui sebuah tabel bilangan integer `DataInt[1..N]`, yang telah diisi. Buatlah sebuah program, untuk mengisi nilai pada tabel dan menghasilkan nilai maksimum dari elemen tabel tersebut:

Ilustrasi :

$N = 8$, `DataInt` berisi : { 4, 2, -5, 10, 11, -6, 3, 10}
Output : Maximum adalah 11

Soal tambahan

- ▶ Nama array adalah DataInt : array [1..50] of integer
- ▶ Buat procedure HitungJumlah dari seluruh isi tabel
- ▶ Buat procedure/ function HitungRata-rata dari seluruh isi tabel, lakukan *call* terhadap procedure HitungJumlah
- ▶ Buat procedure MencariNilaiMinimum dari seluruh isi tabel

Program Dengan Tabel

Contoh 9.7

Detail program utama

Program Tabel

Kamus

Type TabelInt : array[1..100] of integer

Tab : TabelInt

posmax,Max,N : integer

Procedure IsiData(**input** N:integer **I/O** Tab:TabelInt)

Procedure CariMax (**input** X:TabelInt,N:integer **output** max:integer)

Procedure CariPosMax (Input X:TabelInt,N:integer output posmax:integer)

Algoritma

input(N) {N adalah jumlah data pada tabel}

IsiData(N,Tab)

CariMax(Tab,N,max)

CariPosMax(Tab,N,posmax)

output(max); ~~output(X[posmax])~~output(Tab[posMax])

Program Dengan Tabel

Contoh 9.7

Detail prosedur IsiData

```
Procedure IsiData(input N:integer, I/O Tab:TabelInt)
{Mengisi elemen tabel Tab[1..N]dengan harga dari piranti masukan.
IS : Terdefinisi tabel Tab
FS : seluruh elemen tabel berisi harga yang dibaca dari piranti masukan}
```

Kamus Lokal

i : integer

Algoritma

```
For i  $\leftarrow$  1 to N do
    Input(Tab[i])
```

Program Dengan Tabel

Contoh 9.7

Detail dari prosedur CariMax : mencari nilai maks

```
Procedure CariMax(input X:TabelInt,N:integer, output max:integer)
```

```
{Diketahui tabel int X, akan dicari nilai maximumnya pada tabel
```

```
IS : Terdefinisi X
```

```
FS : Dihasilkan nilai maksimum dari X}
```

Kamus Lokal

```
i : integer
```

Algoritma

```
Max ← X[1]
```

```
i ← 2
```

```
While i ≤ N do
```

```
    If X[i] > max then
```

```
        Max ← X[i]
```

```
    i ← i + 1
```

```
{i > N}
```

Program Dengan Tabel

Tabel Tracing

Misal $N = 5$, dan $X = \{20, 12, 32, 5, 40\}$

i	X[i]	X[i] > max	max
			20
2	12	False	20
3	32	True	32
4	5	False	32
5	40	True	40

Program Dengan Tabel

Contoh 9.7

Detail dari prosedur CariPosMax : mencari posisi nilai maks (Seringkali dalam suatu proses, kita membutuhkan indeks di mana nilai maksimum tersebut berada (terutama jika nilai maksimum muncul beberapa kali).

```
Procedure CariPosMax (Input X:TabelInt,N:integer, output posmax:integer)  
{IS : Terdefinisi X bertipe TabelInt}  
{FS : Dihasilkan posisi nilai maksimum dari X}  
Kamus Lokal  
  i : integer  
Algoritma  
  posmax ← 1  
  i ← 2  
  While i <= N do  
    If X[i] > X[posmax] then  
      posMax ← i  
    i ← i + 1  
  {i > N}
```


Program Dengan Tabel

Tabel Tracing

Misal $N = 5$, dan $\text{DataInt} = \{20, 12, 32, 5, 40\}$

i	X[i]	X[i] > X[posmax]	posmax	X[posmax]
			1	20
2	12	False	1	20
3	32	True	3	32
4	5	False	3	32
5	40	True	5	40

- Algoritma ini tidak menghasilkan nilai maksimum, namun karena indeks di mana nilai maksimum tersebut berada diketahui maka nilai maksimum dapat diakses.
- Jika bekerja dengan tabel, lebih disukai bekerja dengan indeks karena berkat indeks nilai dari elemen dapat diakses secara langsung.
- Dengan hanya mengetahui suatu nilai, tidak sederhana untuk mengetahui indeks dimana nilai tersebut berada, karena harus dilakukan dengan proses pencarian

Program Dengan Tabel

Contoh 9.8

Buatlah program untuk menghasilkan nilai rata-rata dari nilai-nilai yang ada pada tabel X. Sebagai ilustrasi :

$$N = 6$$

$$X : \{12, 10, 4, 5, 2, 1\}$$

$$\text{Rata-rata} = 5.67$$

Dalam hal ini akan digunakan function untuk menghasilkan nilai rata-rata. Anda bisa juga menggunakan procedure untuk mencari nilai rata-rata

Program Dengan Tabel

Contoh 9.8

Detail dari program utama

Program NilaiRataRata

Kamus Global

Type TabelInt : array[1..100] of integer

Tab : TabelInt

N : integer {jumlah data pada tabel}

Procedure IsiData(input N:integer, input/output X:TabelInt)

Function rata2(data:TabelInt, N: integer) → Real

Algoritma

Input (N)

IsiData (N, Tab)

output ('rata2= ', rata2 (Tab, N))

Program Dengan Tabel

Contoh 9.8

Detail dari fungsi rata-rata

```
Function rata2 (data:TabelInt; N:integer) → real
```

Kamus

```
  i, Sum : integer
```

Algoritma

```
  Sum ← 0
```

```
  For i ← 1 to N do
```

```
    Sum ← Sum + data[i]
```

```
  → (Sum/N)
```

Program Dengan Tabel

Tabel Tracing

Misal $N = 5$, dan $Data = \{20, 12, 32, 5, 40\}$

i	Data[i]	Sum	Return value
		0	
1	20	20	
2	12	32	
3	32	64	
4	5	69	
5	40	109	$109/5 = 21.8$

Program Dengan Tabel

Contoh 9.9

Buatlah program untuk menghitung jumlah mahasiswa yang nilainya di atas rata-rata. Misal data nilai dari beberapa mahasiswa disimpan dalam tabel. Satu nilai merepresentasikan satu nilai dari satu mahasiswa.

Dalam program ini digunakan prosedur `IsiData` dan prosedur `NilaiAtas`

Program Dengan Tabel

Contoh 9.9

Detail program utama

Program NilaiAtas

Kamus Global

Type TabelInt : array[1..100] of integer

Tab : TabelInt

N : integer {jumlah data pada tabel}

Procedure IsiData(input N:integer, input/output X:TabelInt)

Procedure NilaiAtas(input X:TabelInt, N:Integer, output jum_atas:integer)

Function rata2(X:TabelInt, N) → Real

jumlah : integer

Algoritma

Input (N)

IsiData(N, Tab)

NilaiAtas(Tab, N, jumlah)

output (`jumlah mahasiswa dengan nilai di atas rata-rata =`, jumlah)

Program Dengan Tabel

Contoh 9.9

Detail prosedur NilaiAtas

Procedure NilaiAtas(input X:TabelInt;N:Integer, output jum_atas:integer)

{IS : Terdefinisi TabelInt X yang tidak kosong

FS : Output adalah jumlah mahasiswa yang nilainya di atas rata-rata}

Kamus

i : integer

Function rata2(X:TabelInt, N) → Real

Algoritma

i ← 1

While i ≤ N do

if X[i] > rata2(X, N) then

 jum_atas ← jum_atas + 1

 i ← i + 1

{i > N}

Program Dengan Tabel

Contoh 9.10

Buatlah program untuk mencetak nilai rata-rata serta standar deviasi dari suatu data nilai. Rumus standar deviasi adalah sebagai berikut:

$$s = \sqrt{\sum_{i=1}^n \frac{(x_i - \bar{x})^2}{n-1}}$$

s = standar deviasi

\bar{x} = rata-rata

n = jumlah data

x_i = data ke- i

Program Dengan Tabel

Contoh 9.10

Detail program utama

Program StandarDeviasi

Kamus Global

Type TabelInt : array[1..100] of integer

Tab : TabelInt

N : integer {jumlah data pada tabel}

Procedure IsiData(input N:integer input/output X:TabelInt)

Procedure SDeviasi(input X:TabelInt, N:integer, output s:real)

Function rata2(data:TabelInt, N) → Real

dev : real {Standar Deviasi}

Algoritma

Input (N)

IsiDataInt (N, Tab)

Output ('Nilai rata-rata = ', rata2 (Tab, N))

Sdeviasi (Tab, N, dev)

Output ('Standar Deviasi =, dev)

Program Dengan Tabel

Contoh 9.10

Detail prosedur Sdeviasi (asumsi : terdapat operator kuadrat (SQR) dan Akar(SQRT))

```
Procedure SDeviasi(input X:TabelInt; N:integer, output s:real)  
{IS : Terdefinisi TabInt X yang tidak kosong  
  FS : menghasilkan s yang merupakan nilai standar deviasi data X}
```

Kamus Lokal

```
  nilrata, KuadS, KuadSTemp : real;  
  i : integer
```

Algoritma

```
  nilrata ← rata2(X,N)  
  while i<=N do  
    KuadSTemp ← SQR(X[i]-nilrata)  
    KuadS ← KuadS+KuadSTemp  
    i ← i+1  
{i > N}  
  s ← SQRT (KuadS/(N-1))
```

Program Dengan Tabel 2 Dimensi

Deklarasi

Untuk selanjutnya akan kita lihat tabel 2 dimensi.

Kamus:

TabDua : array[1..100,1..50] of integer

Contoh assignmet:

TabDua[1,5] ← 40

Contoh tabel dua dimensi yang sering kita temui adalah matriks

Program Dengan Tabel 2 Dimensi

Contoh 9.11

Akan dibuat program yang berkaitan dengan proses

- ▶ Pengisian nilai matriks bujur sangkar
- ▶ Pencetakan nilai matriks bujur sangkar
- ▶ Penjumlahan dua buah matrik bujur sangkar
- ▶ Perkalian matrik dengan skalar
- ▶ Perkalian dua buah matrik bujur sangkar

Program Dengan Tabel 2 Dimensi

Contoh 9.11 (Program Utama)

Program OperasiMatriks

Kamus Global

Const baris : integer = 5 {Matrik Bujur sangkar}

Const kolom : integer = 5

Type Matrik : array[1..baris,1..kolom] of integer

MatA,MatB,MatKali,MatTambah : Matrik

Procedure IsiMatrik(input/output Mat:Matrik)

Procedure KaliMatrik(input Mat1, Mat2:Matrik; output MatHasil:Matrik)

Procedure TambahMatrik(input Mat1, Mat2:Matrik; Output MatHasil:Matrik)

Procedure CetakMatrik(input Mat:Matrik)

Algoritma

IsiMatrik(MatA)

IsiMatrik(MatB)

KaliMatrik(MatA,MatB,MatKali)

CetakMatrik(MatKali) {MatKali=matrik hasil kali}

TambahMatrik(MatA,MatB,MatTambah)

CetakMatrik(MatTambah) {MatTambah=matrik hasil penjumlahan}

Program Dengan Tabel 2 Dimensi

Contoh 9.11

Detail prosedur IsiMatriks

```
Procedure IsiMatrik(input/output Mat:Matrik)
{IS : Terdefinisi Matrik Mat yang masih kosong
  FS : Matrik Mat semua elemennya terisi nilai}
Kamus Lokal
  i,j : integer
Algoritma
  for i←1 to baris do
    for j←1 to kolom do
      input(Mat[i,j])
```

Program Dengan Tabel 2 Dimensi

Contoh 9.11

Detail prosedur KaliMatriks

Procedure KaliMatrik(input Mat1, Mat2:Matrik; output MatHasil:Matrik)

{IS : Terdefinisi Matrik Mat1 dan Mat2

FS : MatHasil = Mat1 x Mat2}

Kamus Lokal

i, j, k : integer

Algoritma

for i ← 1 to baris do

for j ← 1 to kolom do

 MatHasil[i, j] ← 0

for k ← 1 to baris do

 MatHasil[i, j] ← MatHasil[i, j] + Mat1[i, k] * Mat2[k, j]

Program Dengan Tabel 2 Dimensi

Contoh 9.11

Detail prosedur TambahMatriks

Procedure TambahMatrik(input Mat1, Mat2:Matrik; output MatHasil:Matrik)

{IS : Terdefinisi Matrik Mat1 dan Mat2

FS : MatHasil = Mat1 + Mat2}

Kamus Lokal

i, j : integer

Algoritma

for i ← 1 to baris do

for j ← 1 to kolom do

 MatHasil[i, j] ← Mat1[i, j] + Mat2[i, j]

Program Dengan Tabel 2 Dimensi

Contoh 9.11

Detail prosedur CetakMatriks

Procedure CetakMatrik(input Mat: Matrik)

{IS : Terdefinisi Matrik Mat

FS : Mencetak elemen dari Matrik Mat}

Kamus Lokal

i, j : integer

Algoritma

for i ← 1 to baris do

for j ← 1 to kolom do

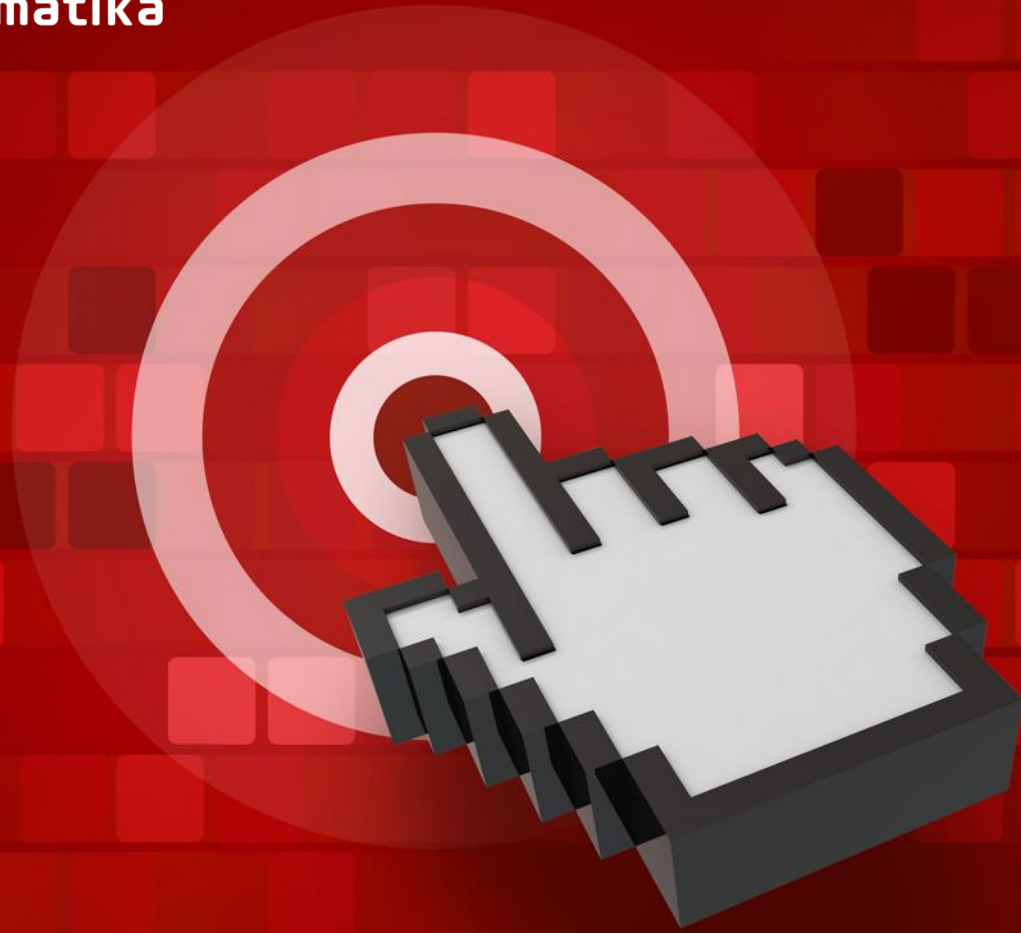
Output(Mat[i, j])

Referensi

- ▶ Inggriani Liem, Diktat Kuliah IF223 Algoritma Dan Pemrograman, Jurusan Teknik Informatika Bandung, 1999



Fakultas Informatika
School of Computing
Telkom University



THANK YOU